Compensation Map Calibration of Engine Management Systems Using Least-Squares Support Vector Committee Machine and Evolutionary Optimization

Pak Kin Wong

Department of Electromechanical Engineering, University of Macau, Taipa, Macau S. A. R, China
*fstpkw@umac.mo

Abstract

Nowadays, automotive engine compensation control is carried out electronically by utilizing many compensation maps in engine management systems; such that the engine can sustain its performance under the variations in engine operating conditions and environmental parameters. In traditional engine compensation map calibration, the parameters are normally set by a trial and error method because the exact mathematical engine model has not been derived. In this paper, a new framework, namely multi-input/output least-squares support vector committee machine, is proposed to construct the engine compensation control system (ECCS) models based on experimental data. As the number of adjustable parameters involved in the ECCS is very huge, the model accuracy and training time are usually degraded. Nonlinear regression is therefore employed to perform dimension reduction before modelling. The ECCS models are then embedded in an objective function for parameter optimization. Two widely-used evolutionary optimization algorithms, Genetic algorithm (GA) and particle swarm optimization (PSO), are applied to the objective function to determine the optimal calibration maps automatically. Experimental results show that the proposed modelling and optimization framework is effective and PSO is superior to the GA in compensation map calibration.

Keywords

EMS Calibration; Compensation Control; Least-Squares Support Vector Machines; Evolutionary Optimization; Committee Machine

Introduction

Nowadays, the automotive engine is controlled by the engine management system (EMS). The EMS calibration is achieved by parameterizing EMS software and looking for an optimal engine performance. The parameters stored in the EMS are organized in a map format. Calibrated maps under normal operating temperatures and conditions are called base maps. The base maps at least include base fuel map and base ignition map. Although the base map represents the fine-tuned performance of a vehicle engine, its performance also changes due to different engine operating and environmental conditions, such as barometric pressure (BP), engine temperature (ET), inlet air temperature (IAT) and battery voltage (BV).

In order to maintain the engine performance under different environmental and engine operating conditions, every EMS has to include an engine compensation control system (ECCS) which usually organizes all control parameters in many compensation maps. The ECCS picks the trim values from the compensation maps based on the signals reported from the IAT, ET, knock and oxygen sensors. The trim values are then added to or deducted from the fuel injection duration and ignition timing of the base fuel and ignition maps to become the final engine control output. This process provides an adaptive engine control action. In addition, the ECCS should essentially allow the engine to achieve rapid warm-up, good acceleration and deceleration responses, fair knock intensity, short cranking time (CT), sustainable air-ratio (lambda value) and torque under various weather and operating conditions.

In engine compensation control (ECC), a compensation map is represented as a row of trim values that can normally be visualized in form of curves. There are many compensation maps in an ECCS, including (1) the fuel compensation maps for IAT, ET, BP, BV and cranking; (2) the ignition compensation maps for IAT, ET. An example of fuel

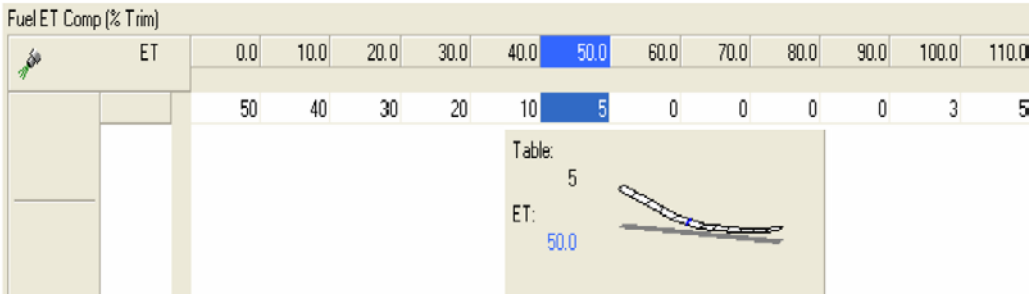

FIG. 1 AN EXAMPLE OF FUEL COMPENSATION MAP FOR ET WHERE THE ET IS DISCRETELY DIVIDED

compensation map and curve for ET is shown in Fig. 1 (MoTeC, 2002); where the engine temperature is discretely divided into $0^{o}C$,$10^{o}C$,...,$110^{o}C$, and the values in the table represent the percentages of fuel trims, which are tunable parameters. In practice, all compensation curves or trim values are engine dependent. Every car model has to undergo a complicated calibration process of compensation maps during development.

The ultimate goal of the engine compensation map calibration (ECMC) is to tune up the engine optimally under different environmental and engine conditions but the goal is extremely difficult to achieve. Current practice of ECMC relies on the experience of the automotive engineer who handles a huge number of combinations of engine control parameters. The relationship between the input and output variables of a modern car engine is a complicated multi-variable nonlinear function (Robert Bosch, 2004; Li, 2007) because the automotive engine is an integrated system of thermo-fluid, electromechanical and computer control systems. Consequently, ECMC is usually performed empirically by the automotive engineer through many experiments (Robert Bosch, 2004; Li, 2007). Therefore, vehicle manufacturers normally spend up to 12 months to tune up an EMS optimally for a new car model (Bell, 2006).

To date, a few studies on computer-aided calibration for some simple EMS parameters are available. However, research on computer-aided calibration for engine compensation maps is hardly found in the open literature. The most common calibration approach for simple EMS parameters is simulation-based calibration (Rask & Sellnau, 2004; Tan *et al.*, 2006)), which divides the calibration process into two parts: modeling and optimization. In terms of modeling, a mathematical engine model can be constructed either from empirical models (Saerens *et al.*, 2009; Zhang *et al.*, 2010; Errico *et al.*, 2011; Qi *et al.*, 2011) or commercial engine simulation software such as Engine Analyzer Pro, GT Power and WAVE's 1-D cycle simulation, etc (Wu *et al.*, 2007). The empirical models and simulation software are derived from some simple physical laws. These mathematical models can simulate engine performance approximately, and thus can be employed as objective functions. The optimal EMS parameters can then be determined based on the objective functions by using computer-aided optimization methods such as gradient search methods (Wu *et al.*, 2007) and genetic algorithm (GA) (Errico *et al.*, 2011). Hence, the amount of time and resources required for engine development can be significantly reduced. However, the predictive accuracy of the mathematical engine models is poor due to the following facts:

1. To predict engine performance, the simulation software and mathematical models require the engineer to provide many engine-specific parameters such as combustion efficiency. In practice, it is too demanding to obtain or estimate such parameters.
2. Excessive numbers of assumptions have been made in these models. This renders such engine models overly simple in comparison with real engine systems.

Several recent studies have described the use of neural networks (NNs) to process collected experimental data and generate engine performance models (Celik & Arcaklioglu, 2005; Garcia-Nieto *et al.*, 2008; Dickinson and Shenton, 2009; Togun and Baysec, 2010). In this way, many direct search techniques can then be applied to search for the optimal engine parameters automatically. However, there are two main drawbacks to the NN modeling approach:

1. The architecture, including the number of hidden neurons, must be determined a priori or modified during training by heuristics, which results in a sub-optimal network structure.
2. During the training process, NNs can easily

become stuck in local minima. Various methods of avoiding local minima (e.g. early stopping and weight decay) have been employed. However, these methods greatly affect the generalization of the estimated function (Haykin, 1999).

Recently, an emerging technique of least-squares support vector machines (LS-SVM) (Suykens *et al.*, 2002) has been applied to model the engine performance (Vong *et al.*, 2006). LS-SVM combines the advantages of neural networks (particularly their ability to handle a large amount of highly nonlinear data) and nonlinear regression (with its high capacity for generalization). It can achieve better accuracy than traditional NNs for automotive engine performance modelling. Subsequently, the authors proposed the application of multi-input/output least-squares support vector machines (MIO LS-SVM) and a GA to determine the optimal set-points for a proportional-integral-derivative idle-air valve controller and some engine control variables for automotive engine idle-speed control (Wong, Tam *et al.*, 2008), which is a very difficult multi-objective optimization problem. The integrated approach can overcome all of the limitations of the existing EMS calibration methods mentioned above. However it is not suitable for modelling the ECCS because the input variables of the ECCS model are usually a mixture of tunable and untunable parameters. The untunable parameters are uncontrollable environmental and operating factors such as ET, IAT, BV, and BP, which cannot be directly regulated by the optimization algorithm. Nevertheless, these untunable parameters must be taken into account for obtaining optimal engine performance at different environmental and engine operating conditions. As the MIO LS-SVM+GA methods (Wong, Tam *et al.*, 2008) can deal with tunable parameters only, this paper proposes a new framework utilizing MIO LS-SVM and committee machine (Krogh & Vedelsby, 1995; Haykin 1999; Chen et al., 2009), namely, multi-input/output least-squares support vector committee machine (MIO LS-SVCM) to model the complex ECCS as an objective function, and then proper optimizer can perform optimization based on this kind of objective function.

Proposed Modelling and Optimization Framework

The MIO LS-SVCM framework is inspired from the concept of committee machines that a group of experts (models) can produce a better result relative to a single expert; especially when the training data of an application can be separated independently. However, there is no universal formulation for committee machines, the application of committee machines is case dependent (Haykin, 1999). For this application, the detail formulation is specially designed. The proposed MIO LS-SVCM firstly separates the whole ECCS model into some simpler sub-models according to the feasible combinations of untunable parameters, such as IAT, ET and BV. In this treatment, the sub-models do not contain any untunable parameters that enable the compensation map optimization. In order to restrict the number of combinations of the untunable parameters, the values of the untunable parameters should be bounded and discretely divided. This idea matches with the general compensation map format of the EMS (Fig. 1). As there are still many adjustable parameters for the divided ECC sub-models, the model accuracy and training time are usually degraded. However, the relationships of adjustable parameters in some compensation maps could be expressed as compensation curves, so nonlinear regression can be applied to represent the data points of curves by several coefficients so as to realize dimension reduction before modelling. Afterwards, the multi-input/output LS-SVM can be applied to construct the ECC sub-models individually. Each ECC sub-model represents one engine output performance at a specific combination of engine operating and environmental condition. Finally, the ECC sub-models are weighted and systemically aggregated to form an objective function (i.e. committee machine) for ECMC.

A schematic illustration of the framework and overall methodology of LS-SVCM is shown in Fig. 2. The upper branch in Fig. 2 shows the steps required to construct the MIO LS-SVCM model. Experiments are still required but only to provide sufficient data for LS-SVM training and validation. The design of experiments (DOE) methodology is additionally used to streamline the process of creating representative sampling data points to train the models. Normalization is applied to preprocess the tunable parameters in the experimental datasets. Once the ECC sub-models obtained are validated in terms of accuracy, it becomes possible to use a proper computer-aided optimization technique to search for the best engine calibration maps automatically. The entire model derived by MIO LS-SVCM is obviously discontinuous such that it is non-differentiable, and gradient information of this model cannot be obtained. In practice, evolutionary optimization methods are usually employed to determine the optimal solution

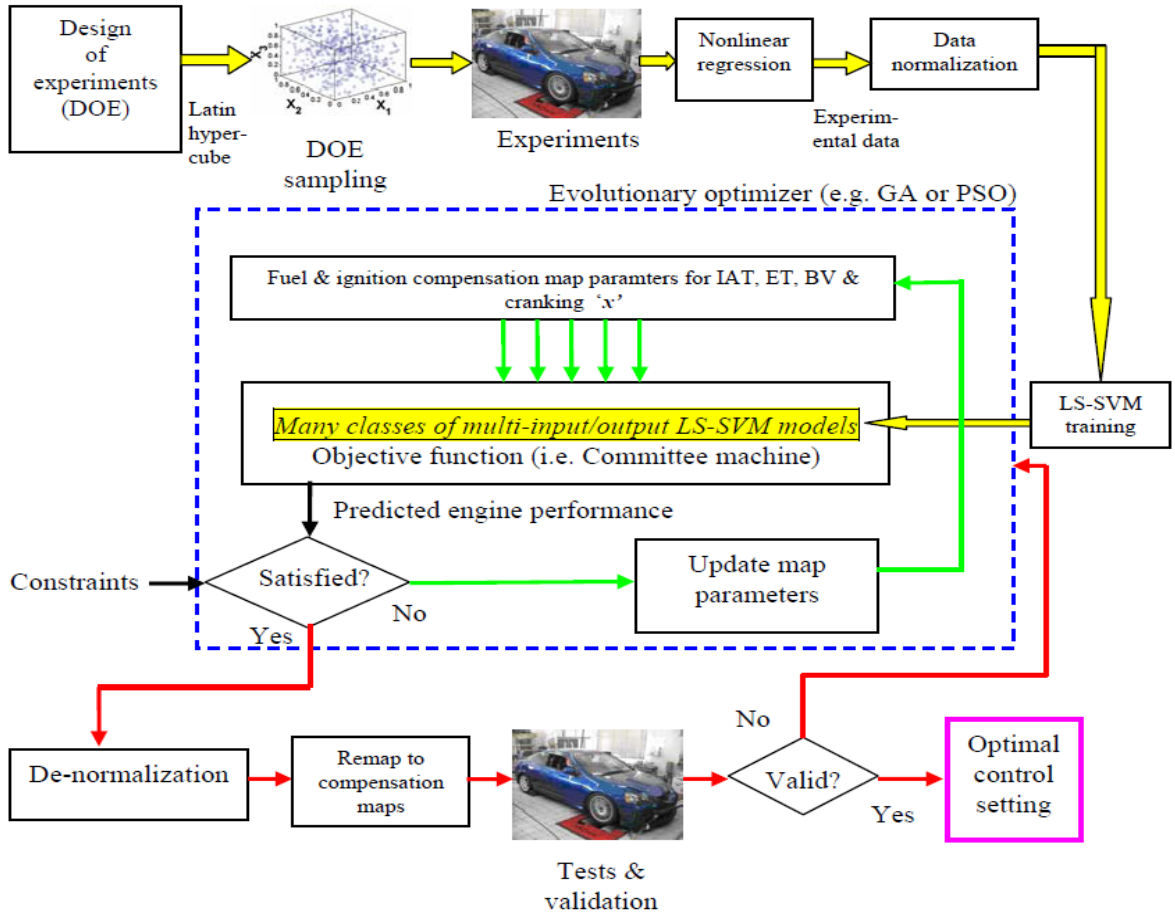

FIG. 2 FRAMEWORK FOR MODELING AND OPTIMIZATION OF ECC PARAMETERS

for this kind of discontinuous problem because no gradient information of the model is required. Evolutionary optimization in general is methods that simulate natural evolution for the task of global optimization. However, there are many evolutionary optimization algorithms available in the open literature, it is impossible to exam all of them. In order to check if the proposed framework can fit various evolutionary optimization algorithms, two widely-used evolutionary optimization techniques, genetic algorithm (GA) and particle swarm optimization (PSO), were selected and tested for this large-scale constrained multivariable optimization problem. GA has already been a well-known technique for solving many automotive engineering optimization problems (Chiou & Liu, 2009; Gauchia et al., 2010). PSO (Kennedy & Eberhart, 1995) is also an evolutionary computation technique similar to the GA. However, it has no evolutionary operators of crossover and mutation as with the GA. From the user point of view, PSO can reduce the time and burden of trying different evolutionary operators to find the optimal solution.

In the lower branch of Fig. 2, the optimal ECC set-points by both optimizers are denormalized into true values and remapped into compensation maps according to the format of the EMS. Finally, the true set-points in the maps are loaded to the EMS to carry out evaluation tests and comparison. The feasibility and efficiency of the two optimization approaches can therefore be examined. The purpose of this paper is to demonstrate the effectiveness of the proposed MIO LS-SVCM + GA/PSO methods on ECMC problems. The detail operation of the modelling and optimization framework and the specific techniques are described in the following sections.

Engine Compensation Control Model Identification and Case Study

LS-SVM Formulation for Multi-class and Multivariable Function Construction

The typical LS-SVM modelling algorithm is a multi-input and single-output modelling method. To deal with the current ECMC problem, the MIO LS-SVCM model is constructed based on many typical LS-SVM models. The concept is presented below.

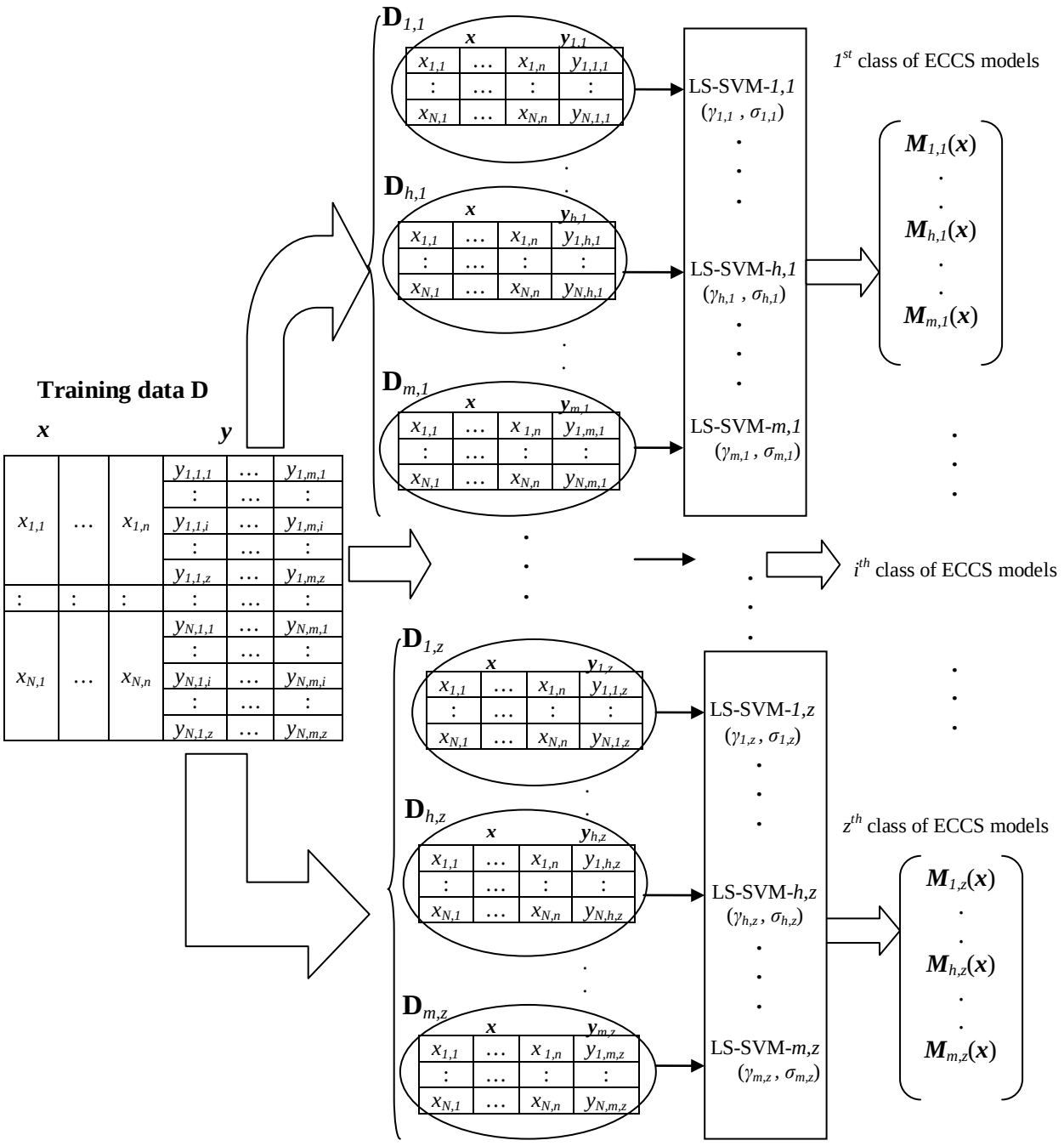

FIG. 3 MIO LS-SVCM MODELLING FRAMEWORK

Consider a training dataset, $\mathbf{D} = \{(\boldsymbol{x}_k, \boldsymbol{y}_{k,i})\}_{k=1,i=1}^{N,z}$, with N data points where $\boldsymbol{x}_k \in R^n$ represents the k^{th} ECC parameter set-up (i.e. set-points), and $\boldsymbol{y}_{k,i} \in R^m$ is the k^{th} corresponding engine output performance, k = 1 to N and i = 1 to z. The index i represents a reasonable combintation of untunable environmental parameters or engine operating conditions. In this study, this set of untunable parameters contains the engine operating conditions of ET, IAT, and BV only. For example, when i = 1, the combination of untunable parameters is ET = 20℃, IAT = 20℃, and BV = 12V. Here, $\boldsymbol{y}_{k,i}$ is an m-dimensional output performance vector, $\boldsymbol{y}_{k,i}$= $[y_{k,1,i},...,y_{k,m,i}]$. In this applicaton, $y_{k,1,i}$ could be the engine knock intensity at the i^{th} engine operating condition and $y_{k,m,i}$ could be the cranking time at the i^{th} engine operating condition. For an automotive engine, each engine output performance in $\boldsymbol{y}_{k,i}$ is usually an individual variable and can be measured separately, so the training dataset $\mathbf{D}$ can be arranged independently as follows:

$$\mathbf{D} = \{(\mathbf{D}_{1,1},\dots \mathbf{D}_{h,1},\dots\mathbf{D}_{m,1}),.. (\mathbf{D}_{1,i},\dots \mathbf{D}_{h,i},\dots\mathbf{D}_{m,i}),.. (\mathbf{D}_{1,z},\dots \mathbf{D}_{h,z},\dots\mathbf{D}_{m,z})\}, \quad (1)$$

where $\mathbf{D}_{h,i} = \{(\boldsymbol{x}_k, y_{k,h,i})\}_{k=1,i=1}^{N,z}$; $h \in [1,m]$. In other words, the entire training dataset $\mathbf{D}$ is separated into z classes × m multi-input/single-output sub-training datasets $\mathbf{D}_{h,i}$ (Fig. 3). For each multi-input/single-output dataset $\mathbf{D}_{h,i}$, the LS-SVM dual formulation for nonlinear function estimation as given in Equation (2) (Suykens *et al.*, 2002) can be applied to.

$$\left[\begin{array}{l}\text{Solve in } \boldsymbol{\alpha}_{h,i},\ b_{h,i}: \\ \quad \begin{bmatrix} 0 & \mathbf{1}_v^T \\ \mathbf{1}_v & \boldsymbol{\Omega} + \frac{1}{\gamma_{h,i}} \mathbf{I}_N \end{bmatrix} \begin{bmatrix} b_{h,i} \\ \boldsymbol{\alpha}_{h,i} \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{y}_{h,i} \end{bmatrix}\end{array}\right] \quad (2)$$

Where $\mathbf{I}_N$ is an N-dimensional identity matrix, $\boldsymbol{y}_{h,i} = [y_{1,h,i}, \ldots, y_{N,h,i}]^T$, $\mathbf{1}_v$ is an N-dimensional vector = $[1,\ldots,1]^T$, and $\boldsymbol{\alpha}_{h,i} = [\alpha_{1,h,i},\ldots, \alpha_{N,h,i}]^T$. The resulting LS-SVM model for function estimation becomes:

$$\begin{aligned} M_{h,i}(\boldsymbol{x}) &= \sum_{k=1}^{N} \alpha_{k,h,i} \phi(\boldsymbol{x}_k)^T \phi(\boldsymbol{x}) + b_{h,i} \\ &= \sum_{k=1}^{N} \alpha_{k,h,i} K(\boldsymbol{x}_k, \boldsymbol{x}) + b_{h,i} \\ &= \sum_{k=1}^{N} \alpha_{k,h,i} \exp\left(-\frac{\|\boldsymbol{x}_k - \boldsymbol{x}\|^2}{\sigma_{h,i}^2} \right) + b_{h,i} \quad , \end{aligned} \quad (3)$$

where $\alpha_{k,h,i}$, $b_{h,i} \in R$ are the solutions of Equation (2), $\boldsymbol{x}$ is the new input ECC set-points for engine performance prediction. After inferring $m \times z$ pairs of hyperparameters ($\gamma_{h,i}$, $\sigma_{h,i}$) by Bayesian inference (Vong *et al.*, 2006), $m \times z$ individual training datasets are used to calculate $m \times z$ individual sets of support vectors $\alpha_{k,h,i}$ and threshold values $b_{h,i}$. Finally, $m \times z$ individual sets of $M_{h,i}(\boldsymbol{x})$ are constructed based on Equation (3). Each $M_{h,i}(\boldsymbol{x})$ sub-model is viewed as a committe member in the committee machine. The whole MIO LS-SVCM modelling algorithm is shown in Fig. 3, where a set of typical LS-SVM sub-models are generated to predict the engine output performances at the i^{th} reasonable combination of engine operating condition for different combinations of ECC variables in $\boldsymbol{x}$.

Experimental Set-up for Case Study

In this case study, a Honda Integra DC5 Type-R with a K20A DOHC i-VTEC engine was used as a test car to demonstrate the feasibility of the proposed methodology through a programmable MoTeC M800 EMS (Fig. 4). As an illustration, six compensation maps were considered in the experiment, including the fuel compensation maps for IAT, ET, CT and BV, as well as ignition compensation maps for IAT and ET. The fuel compensation map for BP was not considered because the base map of the test engine uses speed-density system, which estimates engine load by measuring manifold-absolute pressure. The speed-density system can therefore automatically correct for BP changes (Hartman, 1993). Moreover, four engine output performance data were considered including knock intensity (*knk*), cranking time, differential lambda ($\Delta\lambda$) and differential engine torque (ΔT). The lambda value and knock intensity were measured by a wide-band lambda sensor and a knock monitor respectively. The cranking time was recorded by the built-in data logging function in the MoTeC EMS. The differential engine torque was measured by running the test car on a chassis dynamometer (dyno) at an idle speed and full-load condition at random engine speeds, so that various conditions of ET and IAT were achieved for data sampling. All of the experimental data were collected under a controlled ambient environment using powerful air-conditioners and fans.

Selection of Ranges

In this case study, the fuel and ignition compensation maps are the functions of the untunable environmental parameters IAT, ET, and BV. Owing to the limitations in the automotive laboratory equipment at the University of Macau, test car characteristics, and weather in Macau, the sampling temperature and BV ranges were selected as: 15℃ to 55℃ for IAT, 15℃ to 105℃ for ET and 11.5V to 13.5V for BV. Based on the format of the programmable EMS, the intervals for IAT and ET were set at 10℃, i.e. IAT= (20, 30,…,50), ET = (20, 30,…,100) and for BV, the interval was set at 1, i.e. BV = (12, 13).

According to the concept of MIO LS-SVCM, the complete ECCS model is therefore separated into sub-models according to different temperature and BV intervals. The indexes (*e*, *a*, *b*) were used directly as indexes for ET, IAT and BV respectively. The indexes of *e*, *a*, and *b* covered the range for e ± 5℃, a ± 5℃ and b ± 0.5V respectively. For example, the data from 15℃ to 25℃ were viewed as 20℃, then 20 was used as one of the model indexes.

Originally, the number of combinations of such indexes is 72. However, many combinations of IATs and ETs are unreasonable. For example, it is impossible to have 20℃ IAT and 100℃ ET simultaneously. By considering the real situations, the number of feasible combinations was reduced to 30 (i.e. 15 feasible combinations of IATs and ETs × 2 BV combinations) as shown in Table 1.

FIG. 4 EXPERIMENTAL SET-UP FOR DATA SAMPLING AND PROGRAMMABLE EMS

TABLE 1 FEASIBLE COMBINATIONS OF IATs AND ETs WITH BVs = 12V AND 13V

Index of ET (℃) / Index of IAT (℃)	20	30	40	50	60	70	80	90	100
20	1	1	1	1	1	1	0	0	0
30	0	0	0	1	1	1	1	1	1
40	0	0	0	0	0	0	0	1	1
50	0	0	0	0	0	0	0	0	1

(1: feasible; 0: unreasonable)

TABLE 2 FEASIBLE INPUT VARIABLES FOR THE USER TO FILL IN THE COMPENSATION MAPS

Compensation map	Maps for adjustment											Sub-total:
Fuel-IAT(℃)	10	20	30	40	50	60						6
Fuel-ET(℃)	10	20	30	40	50	60	70	80	90	100	110	11
Ignition-IAT(℃)	10	20	30	40	50	60						6
Ignition-ET(℃)	10	20	30	40	50	60	70	80	90	100	110	11
Fuel-CT(μsec)	1 variable											1
Fuel-BV(V)	11	12	13	14								4
											Total:	39

Dimension Reduction of Input Variables by Nonlinear Regression

According to the selected maps, ranges, intervals, indexes and the format of the EMS, there were six compensation maps with 39 input variables (Table 2). However, 39 variables are still too much. To avoid handling more of such variables, each fuel and ignition map for IAT and ET were converted into compensation curves with 4 coefficients, which are a_u, b_u, c_u, d_u with u =1,2,3,4 , using nonlinear regression. For each curve, the x-axis represents the value of IAT and ET, and the y-axis represents the value of compensation factors of fuel or ignition. The compensation factors for fuel and ignition are percentages of fuel trim and degrees of ignition angle trim respectively. One more digit, $p_u \in \{0, 1\}$; $u \in [1,4]$, was used as identity digit to represent the categories of curves for IAT and ET. By referring to (Hartman, 1993; MoTeC, 2002; Wang *et al*., 2006, Wong, Vong *et al*., 2008), the compensation curves for cranking and BV can also be expressed in linear and hyperbolic functions respectively. Therefore, the input $\boldsymbol{x}$ was reduced from 39 to 23 as shown in Equation (4). The parameters in the vector $\boldsymbol{x}$ are tunable/adjustable parameters.

$$\boldsymbol{x} = < a_1, b_1, c_1, d_1, p_1, a_2, b_2, c_2, d_2, p_2, a_3, b_3, c_3, d_3, p_3, a_4, b_4,$$

c_4, d_4, p_4, a_5, a_6, b_6 > (4)

where

a_1, b_1, c_1, d_1: Coefficients of fuel curve for IAT compensation

p_1: Identity digit of fuel curve for IAT compensation

a_2, b_2, c_2, d_2: Coefficients of fuel curve for ET compensation

p_2: Identity digit of fuel curve for ET compensation

a_3, b_3, c_3, d_3: Coefficients of ignition curve for IAT compensation

p_3: Identity digit of ignition curve for IAT compensation

a_4, b_4, c_4, d_4: Coefficients of ignition curve for ET compensation

p_4: Identity digit of ignition curve for ET compensation

a_5: Coefficients of fuel line for cranking compensation

a_6, b_6: Coefficients of fuel curve for BV compensation

Overall, the shape of the compensation curves could generally be represented as a third order polynomial, Equation (5), or sinusoidal function, Equation (6), or hyperbolic function, Equation (7) according to many text books and user manuals (Hartman, 1993; MoTeC, 2002; Wang *et al*., 2006). Linear regression can also be obtained using the polynomial regression of Equation (5) by setting the first two coefficients to zero.

$$Y = a_u X^3 + b_u X^2 + c_u X + d_u \quad (5)$$

$$Y = a_u + b_u(\cos(c_u X) + d_u) \quad (6)$$

$$Y = \frac{a_6}{X} + b_6 \quad (7)$$

Where X and Y are the map indexes (e.g. at 30℃ ET, X = 30) and compensation factor (e.g. for -4.5% of fuel trim, Y = -4.5) in the compensation maps respectively. In the input vector $\boldsymbol{x}$, Equations (5) and (6) are respectively represented by setting p_u to be 0 and 1 for each u. The use of the digital number as an extra tunable parameter allows the computer optimizers to evaluate and determine the best shape of compensation curve in the course of optimization.

TABLE 3 DIVISION OF DATASET **D** INTO 30 SUB-DATASETS $\mathbf{D}_{h,e,a,b}$ ACCORDING TO VARIOUS ETs, IATs and BVs

									***h*=1**	***h*=2**	***h*=3**	***h*=4**
i	***k***	$\mathbf{D}_{h,e,a,b}$	a_1	b_1	…	p_1	…	b_6	Knk_i **(%)**	CT_i **(s)**	$\Delta\lambda_i$	ΔT_i **(Nm)**
1	1	$\mathbf{D}_{h,20,20,12}$	5.1428	-0.2587	…	0	…	0.2345	15.8	2.8	0.092	3.7
2	1	$\mathbf{D}_{h,30,20,12}$	5.1428	-0.2587	…	0	…	0.2345	15.9	5.3	0.078	3.1
⋮	⋮	⋮	⋮	⋮	…	⋮	…	⋮	⋮	⋮	⋮	⋮
30	1	$\mathbf{D}_{h,100,50,13}$	5.1428	-0.2587	…	0	…	0.2345	15.7	2.7	0.088	-10.7
1	2	$\mathbf{D}_{h,20,20,12}$	7.7142	-0.4607	…	1	…	0.2675	15.8	4.9	0.078	3.1
2	2	$\mathbf{D}_{h,30,20,12}$	7.7142	-0.4607	…	1	…	0.2675	15.8	3.4	0.225	-3.2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
30	2	$\mathbf{D}_{h,100,50,13}$	7.7142	-0.4607	…	1	…	0.2675	15.6	3.4	0.027	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	300	$\mathbf{D}_{h,20,20,12}$	9.1428	-0.4964	…	1	…	0.2126	15.8	3.3	0.225	3.2
2	300	$\mathbf{D}_{h,30,20,12}$	9.1428	-0.4964	…	1	…	0.2126	16.4	3.2	0.153	-3.5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
30	300	$\mathbf{D}_{h,100,50,13}$	9.1428	-0.4964	…	1	…	0.2126	16.1	1.6	0.09	-8.4

Data Sampling and Division for Case Study

In view of the structure of MIO LS-SVCM, the data was sampled and divided according to different feasible combinations of IATs, ETs and BVs. Moreover, the training dataset was expressed as: $\mathbf{D}_{h,i}=\mathbf{D}_{h,e,a,b}=\{\boldsymbol{x}_k, \boldsymbol{y}_{k,i}\}=\{\boldsymbol{x}_k, \boldsymbol{y}_{k,e,a,b}\}=\{\boldsymbol{x}_k, y_{k,h,e,a,b}\}$, where k = 1 to 300 in this case study. The index i represents the i^{th} combination of (e, a, b). $y_{k,h,e,a,b}$ is the h^{th} engine performance data in the k^{th} sample data point at e, a and b. Specifically, $\boldsymbol{y}_{k,i}$ is defined as follows:

$$\boldsymbol{y}_{k,i} = \{knk_i, CT_i, \Delta\lambda_i, \Delta T_i\} = \{knk_{e,a,b}, CT_{e,a,b}, \Delta\lambda_{e,a,b}, \Delta T_{e,a,b}\} \quad (8)$$

and

$$\Delta\lambda_i = |\lambda_i - \lambda'| \quad (9)$$

$$\Delta T_i = T_i - T' \quad (10)$$

For the above datasets, knk_i is the engine knock intensity, CT_i is the engine cranking time, $\Delta\lambda_i$ is the error between the lambda value produced by the base maps and the lambda value after compensation, ΔT_i is the error between the output torque produced by the base maps and the torque after compensation. λ' and T' are respectively the lambda value and the output torque at normal operating temperatures e = 90℃, a = 40℃ and b = 13V without compensation. According to Table 1, the feasible combination of the training data and LS-SVM models are shown below:

i^{th} *combination* = (e,a,b) =

[20,20,12 30,20,12 40,20,12 50,20,12 60,20,12 70,20,12
50,30,12 60,30,12 70,30,12 80,30,12 90,30,12 100,30,12
90,40,12 100,40,12 100,50,12 20,20,13 30,20,13 40,20,13
50,20,13 60,20,13 70,20,13 50,30,13 60,30,13 70,30,13
80,30,13 90,30,13 100,30,13 90,40,13 100,40,13 100,50,13] (11)

In the above representation, when i=1, (e,a,b) = (20,20,12); i=2, (e,a,b) = (30,20,12);……; i=30, (e,a,b) = (100,50,13). According to above combination, **D** could be grouped into 30 training model datasets, namely $\mathbf{D}_{h,20,20,12}$, $\mathbf{D}_{h,30,20,12}$, …, $\mathbf{D}_{h,100,50,13}$. The training dataset **D** is expressed in Table 3.

Latin hypercube sampling (Lunani *et al.*, 1995) is a technique of DOE which was employed to select a representative set of operating points to generate training samples. A total of 300 sets of representative combinations of input variables were selected and loaded to the EMS to produce 36000 sets (i.e. 300 data × 30 training model datasets × 4 output performance values) of engine output performance data. After collecting a sample dataset **D**, every data subset $\mathbf{D}_{h,i}\subset\mathbf{D}$ was randomly divided into two sets: $TRAIN_{h,i}$ for training and $TEST_{h,i}$ for testing, such that $\mathbf{D}_{h,i} = TRAIN_{h,i}\cup TEST_{h,i}$, where $TRAIN_{h,i}$ contained 80% of $\mathbf{D}_{h,i}$, $TEST_{h,i}$ held the remaining 20% for h =[1,4], and the engine performance for each h in this case study is shown in Table 3. Subsequently every $TRAIN_{h,i}$ was sent to the LS-SVM program for training which was implemented using MATLAB on a PC.

Data Normalization

To obtain a more accurate modelling result, the input data of the training dataset is conventionally normalized before training (Pyle, 1999). This procedure prevents any input parameter from dominating the output value. For every input parameter, it was normalized to within the range [0, 1], i.e., a unit variance, using Equation (12).

$$v^* = \frac{v - v_{\min}}{v_{\max} - v_{\min}} \quad (12)$$

The limits of $v_{\min}$ and $v_{\max}$ for each variable were obtained from handbooks or expert knowledge. After obtaining the optimal setting, each set-point must be de-normalized to obtain the actual value v. The processes of normalization ($v \rightarrow v^*$) and de-normalization ($v^* \rightarrow v$) were performed by using Equation (12). The process flow of the normalization and de-normalization is shown in Fig. 2.

Modelling Results of MIO LS-SVCM

After data division and pre-processing, each subset $\mathbf{D}_{h,i}$ was passed to the LS-SVM program in Equation (2) in order to construct the function $M_{h,i}(\boldsymbol{x})$ in Equation (3). According to the division of the training data, there were totally 120 output models trained for every h and i (i.e. 4 performance outputs × 30 classes of models). In other words, the LS-SVM program was run for 120 times. In every run, a different subset $TRAIN_{h,i}$ was used as the training dataset to construct the corresponding engine output performance function. The one hundred and twenty trained engine performance functions or models $M_{h,i}(\boldsymbol{x})$ are grouped as follows:

$$[M_{h,i}(\boldsymbol{x})] = [M_{h,e,a,b}(\boldsymbol{x})] = [M_{h,20,20,12}(\boldsymbol{x}) \quad M_{h,30,20,12}(\boldsymbol{x}) \quad \ldots\ldots \quad M_{h,100,50,13}(\boldsymbol{x})] \quad (13)$$

Where h is the index of output performance function of the knock intensity, cranking time, air-ratio difference and engine torque difference, and they are

given in Equations (14) to (17).

For $h=1$, $M_{1,i}(\boldsymbol{x}) = knk_i(\boldsymbol{x})$ (14)

For $h=2$, $M_{2,i}(\boldsymbol{x}) = CT_i(\boldsymbol{x})$ (15)

For $h=3$, $M_{3,i}(\boldsymbol{x}) = \Delta\lambda_i(\boldsymbol{x}) = |\lambda_i(\boldsymbol{x}) - \lambda'|$ (16)

For $h=4$, $M_{4,i}(\boldsymbol{x}) = \Delta T_i(\boldsymbol{x}) = T_i(\boldsymbol{x}) - T'$ (17)

where

$knk_i(\boldsymbol{x}) = knk_{e,a,b}(\boldsymbol{x})$: Knock intensity function for engine temperature e, inlet air temperature a and battery voltage b.

$CT_i(\boldsymbol{x}) = CT_{e,a,b}(\boldsymbol{x})$: Cranking time function for e, a and b.

$\lambda_i(\boldsymbol{x}) = \lambda_{e,a,b}(\boldsymbol{x})$: Lambda function for e, a and b.

$T_i(\boldsymbol{x}) = T_{e,a,b}(\boldsymbol{x})$: Output torque function for e, a and b.

In the aforementioned functions, e,a,b are used as a function index again. For example: when $\lambda_{e,a,b}(\boldsymbol{x}) = \lambda_{80,30,12}(\boldsymbol{x})$, it represents a lambda function for the engine temperature $e \approx 80$℃, inlet air temperature $a \approx 30$℃ and battery voltage $b \approx 12$V. The accuracy of each function of $M_{h,i}(\boldsymbol{x})$ was verified using an error function proposed in Equation (18).

$$E_{h,i} = \sqrt{\frac{1}{N_t}\sum_{t=1}^{N_t}\left[\frac{y_{t,h,i} - M_{h,i}(\boldsymbol{x}_t)}{y_{t,h,i}}\right]^2} \quad (18)$$

where $\boldsymbol{x}_t$ is the ECC parameter set-up of the t^{th} data point in $TEST_{h,i}$, $y_{t,h,i}$ is the actual engine output value of the t^{th} data point $\{\boldsymbol{x}_t, y_{t,h,i}\}$ in $TEST_{h,i}$, and N_t is the number of data points in the test dataset. The error $E_{h,i}$ is a root-mean-square of the difference between the true value $y_{t,h,i}$ and its corresponding value estimated from $M_{h,i}(\boldsymbol{x}_t)$. The difference is also divided by $y_{t,h,i}$, so that the result is normalized to within the range [0,1]. Hence, the accuracy rate for each output function $M_{h,i}(\boldsymbol{x})$ is calculated using Equation (19).

$$Accuracy_{h,i} = (1 - E_{h,i}) \times 100\% \quad (19)$$

TABLE 4 ACCURACIES OF DIFFERENT ENGINE OUTPUT PERFORMANCE FUNCTIONS

i	Engine output function index e,a,b	Mean accuracy of knk_i with $TEST_{1,i}$ (%)	Mean accuracy of CT_i with $TEST_{2,i}$ (%)	Mean accuracy of $\Delta\lambda_i$ with $TEST_{3,i}$ (%)	Mean accuracy of ΔT_i with $TEST_{4,i}$ (%)
1	*20,20,12*	98.97	90.85	92.24	96.77
2	*30,20,12*	99.06	83.36	94.74	96.44
3	*40,20,12*	98.00	92.44	93.36	90.19
4	*50,20,12*	99.04	88.85	98.44	94.17
5	*60,20,12*	98.82	92.87	97.74	92.45
6	*70,20,12*	97.49	88.54	93.72	93.72
7	*50,30,12*	98.08	91.31	94.25	95.30
8	*60,30,12*	98.56	83.03	90.24	96.06
9	*70,30,12*	99.24	85.73	97.20	94.87
10	*80,30,12*	98.56	91.09	91.53	92.81
11	*90,30,12*	97.23	92.73	94.87	91.24
12	*100,30,12*	98.67	91.35	92.81	93.42
13	*90,40,12*	98.34	86.93	91.24	96.67
14	*100,40,12*	97.64	84.38	93.42	97.57
15	*100,50,12*	99.08	86.47	96.67	95.89
16	*20,20,13*	98.37	91.25	97.57	96.63
17	*30,20,13*	98.96	92.70	95.89	93.51
18	*40,20,13*	98.43	86.24	96.63	95.44
19	*50,20,13*	97.24	83.72	93.51	98.36
20	*60,20,13*	98.82	84.90	95.44	96.34
21	*70,20,13*	97.69	87.45	98.36	97.74
22	*50,30,13*	98.38	93.61	96.34	93.72
23	*60,30,13*	98.56	90.46	95.25	94.25
24	*70,30,13*	99.24	93.35	93.14	90.24
25	*80,30,13*	98.56	85.32	99.34	97.20
26	*90,30,13*	98.23	84.89	94.87	91.53
27	*100,30,13*	98.67	91.63	92.81	94.87
28	*90,40,13*	98.34	83.65	91.24	92.81
29	*100,40,13*	97.64	93.12	93.42	91.24
30	*100,50,13*	98.48	90.03	96.67	93.42
Average		98.41	88.74	94.77	94.50
Overall average: 94.10%					

According to the indexes *i* and *h*, each class of engine performance functions contains 7200 (i.e. 4 output performance values × 300 data × 30 classes of models × 20%) test data in $TEST_{h,i}$. Based on the index *h*, each engine performance function uses 1800 (i.e. 7200 data /4 output performance values) test data for evaluation. By comparing $TEST_{h,i}$ with the predicated output performance data from the corresponding engine performance function, the accuracy for each function was obtained as shown in Tables 4. The results indicate that the overall function accuracy is very good, at 94.1%. However, it is believed that the accuracy of the function can be improved by increasing the amount of training data.

Optimization of Compensation Maps

After obtaining the ECCS models by MIO LS-SVM, the models were aggregated to form an objective function. It was then possible to use the GA and PSO to search for the best ECC parameter set-up automatically. In this application, the ECC set-up invloves 23 tunable parameters with multiple optimization objectives and constraints, so it is a challenging large-scale optimization problem. To minimize the chance of obtaining a local optimum, several different and arbitrarily generated initial populations are used. For this purpose, ten independent optimization runs were performed for each optimizaiton method. The results of both optimization algorithms are also compared at the end of this section.

Objective Function for ECMC

An objective function was systemically designed to evaluate the engine performance under different compensation control set-ups. In this case study, the main compensation objective is to minimize the following four basic requirements (Jurgen, 1995) and subject to some constraints:

(1) Fair knock intensity

Higher knock intensity means higher engine torque output. However, high knock intensity decreases the service life of an engine. Empirically, the knock intensity should be adjusted below 60% for common vehicles and the value can be adjusted by the user.

(2) Short cranking time

Under a fine-tuned base map, vehicles are expected to start as rapidly as possible in any weather conditions.

(3) Sustainable air ratio (Ideal value**: $\Delta\lambda=0$)**

The engine compensation control must be capable of maintaining the output lambda value close to the lambda value specified by the base maps under different engine operating conditions.

(4) Sustainable engine torque (Ideal value: $\Delta T \geqq 0$)

This is a balance between the engine torque and knock intensity. When the ignition advance drops down, or becomes a zero value, the knock intensity decreases. However, the engine torque is also decreased significantly. Therefore, a function is required to avoid excessive ignition retard in the ECMC.

According to the above requirements, the objective function for this case study is to minimize the sub-functions from the following equations:

$$f_a(x) = \overline{\Delta\lambda} = \frac{\left|\sum_{i=1}^{30} \lambda_i(\boldsymbol{x}) - \lambda_i^{'}\right|}{30} \tag{20}$$

$$f_b(x) = \overline{CT} = \frac{\sum_{i=1}^{30} CT_i(\boldsymbol{x})}{30} \tag{21}$$

$$f_c(x) = \overline{\Delta T} = \frac{\left[\sum_{i=1}^{30} (T_i(\boldsymbol{x}) - T_i^{'})\right]}{30} \tag{22}$$

Hence, the overall objective function f_{obj} is defined as:

$$f_{obj} = \text{Max}[fitness] = \text{Max}[-W_1\tan^{-1}(f_a(\boldsymbol{x})) - W_2\tan^{-1}(f_b(\boldsymbol{x})) + W_3\tan^{-1}(f_c(\boldsymbol{x}))] \tag{23}$$

W_1, W_2, W_3 are the user-defined weights of $f_a(x)$, $f_b(x)$ and $f_c(x)$ respectively, where $f_a(x)$ is the average air-ratio difference, $f_b(x)$ is the average cranking time and $f_c(x)$ is the average engine torque difference, subject to the following constraint equations:

$$\forall knk_i(\boldsymbol{x}) \leq 60\%,\ i=1,2,\ldots\ldots,30 \tag{24}$$

$$p_u \in \{0,1\} \tag{25}$$

Equation (25) ensures that each selected compensation curve by the optimizer is confined to the shapes of the curves described by Equations (5) to (6). In this case study, W_1, W_2 and W_3 were all set at 1. Each performance index was also transformed to a scale of $(0, \pi/2)$ by $\tan^{-1}(\cdot)$ in Equation (23). This ensures that each index makes the same contribution to the objective function. Since the objective function f_{obj} considers many ECMC requirements and constraints, it is treated as the overall compensation performance of the EMS.

TABLE 5 GA OPERATORS AND PARAMETERS

Number of generation	1000
Population size	50
Selection method	Standard proportional selection
Crossover method	Simple crossover with probability = 80%
Mutation method	Hybrid static Gaussian and uniform mutation with probability = 40% and standard deviation = 0.2

TABLE 6 GA OPTIMIZATION RESULTS OF DIFFERENT OUTPUT FUNCTIONS

Engine output function $M_{h,i}(x)$	$Knk_i(x)$**(%)**	$CT_i(x)$**(s)**	$\Delta\lambda_i(x)$	$\Delta T_i(x)$**(Nm)**
	h=1	*h=2*	*h=3*	*h=4*
$M_{h,20,20,12}(x)$	16.09	0.42	0.41	-0.16
$M_{h,30,20,12}(x)$	16.49	3.38	0.41	1.00
$M_{h,40,20,12}(x)$	16.36	1.87	0.39	6.08
$M_{h,50,20,12}(x)$	16.21	1.46	0.26	2.00
$M_{h,60,20,12}(x)$	16.22	2.08	0.37	1.99
$M_{h,70,20,12}(x)$	16.71	1.83	0.39	0.35
$M_{h,50,30,12}(x)$	16.32	2.62	0.37	0.58
$M_{h,60,30,12}(x)$	16.32	1.41	0.38	-0.05
$M_{h,70,30,12}(x)$	16.32	3.13	0.79	0.27
$M_{h,80,30,12}(x)$	16.09	2.66	0.85	1.77
$M_{h,90,30,12}(x)$	16.67	1.77	1.11	1.59
$M_{h,100,30,12}(x)$	16.60	1.17	0.82	-0.05
$M_{h,90,40,12}(x)$	16.72	2.02	0.38	-0.33
$M_{h,100,40,12}(x)$	16.17	2.78	1.28	0.77
$M_{h,100,50,12}(x)$	16.54	1.34	0.47	0.35
$M_{h,20,20,13}(x)$	16.19	0.45	0.43	-0.21
$M_{h,30,20,13}(x)$	16.27	3.38	0.43	0.80
$M_{h,40,20,13}(x)$	16.46	1.87	0.41	6.28
$M_{h,50,20,13}(x)$	16.31	1.46	0.28	2.20
$M_{h,60,20,13}(x)$	16.32	1.08	0.39	1.79
$M_{h,70,20,13}(x)$	16.81	0.83	0.41	0.55
$M_{h,50,30,13}(x)$	16.42	1.62	0.39	0.78
$M_{h,60,30,13}(x)$	16.42	1.41	0.41	0.14
$M_{h,70,30,13}(x)$	16.42	1.13	0.82	-0.47
$M_{h,80,30,13}(x)$	16.03	2.11	0.64	1.56
$M_{h,90,30,13}(x)$	16.45	1.88	0.47	1.79
$M_{h,100,30,13}(x)$	16.17	3.38	0.43	-1.00
$M_{h,90,40,13}(x)$	16.82	1.02	0.42	-0.12
$M_{h,100,40,13}(x)$	17.03	0.51	0.45	-0.12
$M_{h,100,50,13}(x)$	16.64	0.34	0.49	0.55
Overall average	16.42	1.75	0.52	1.02
Fitness value		-39.5		

GA Setting and Optimization Results

The GA optimization framework was implemented using MATLAB. By testing various crossover and mutation methods for this application, the GA operators and parameters as shown in Table 5 were selected to ensure maximum efficiency and accuracy. Ten sets of optimization results were obtained by the GA, which were different from each other. The standard deviation of the fitness of the ten independent runs is about 0.06. Table 6 shows that the best fitness value out of the ten independent runs is -39.5. The optimized set-points recommended by the GA are then de-normalized into true values and remapped into six compensation maps according to the format of the EMS and Equations (5) to (7). The optimal maps are shown in Table 7.

PSO Setting and Results

The PSO framework was again implemented using MATLAB. In PSO, the population is called a swarm, and the individuals (i.e. different combinations of ECC set-points $\boldsymbol{x}$) are called particles. The selection of the PSO parameters has been widely studied in the related literature (Clerc & Kennedy, 2002; Trelea, 2003). Based on this literature, the PSO parameters as shown in Table 8 were selected for this case study. Ten sets of optimization results were obtained by PSO. The standard deviation of the fitness of the ten independent runs is about 0.02. Table 9 shows the best fitness value of PSO out of the ten independent runs is – 33.7.

The optimized set-points recommended by PSO are also denormalized into true values and reorganized into six compensation maps. The optimal maps are shown in Table 10.

Evaluation of the Results

By comparing the fitness value of PSO with the GA, the fitness value of PSO (-33.7) is larger than that of the GA (-39.5). Therefore, the overall compensation performance using the set-points produced by PSO is better than that based on the GA. By comparing the standard deviation of the ten independent runs of PSO with the GA, PSO is less than the GA. In other words, the PSO results are more stable than those with the GA because PSO is less sensitive to the initial values. Another issue is about the computational time. The computational times for MIO LS-SVCM + GA and MIO LS-SVCM + PSO are almost the same. On a 2.66 GHz Intel Core I7-920 PC with 6 GB RAM on-board, both methods spent approximately 76 hours for modelling and optimization. This result reveals that both computer-aided methods can greatly shorten the calibration time than the traditional manual approach that usually spends many months. Although the computational time of PSO is no advantage over the GA, PSO can provide the maximum fitness and the minimum standard deviation of the fitness. Hence, PSO is recommended to be the optimization tool for this problem domain. The aforementioned results show that PSO is superior to the GA in this application. The reasons for this superiority may be that the PSO mechanism generates a more diverse population during the whole iteration process.

The actual output performance using the set-points recommended by PSO was also loaded to the test car and evaluation tests were performed using the chassis dyno. The overall accuracy of the optimal output performance 'O_p' relative to actual test results 'D_a' is 94.2%, again verifying that the ECCS models using the MIO LS-SVCM method are accurate and reliable. The overall actual improvement of D_a over D_{best}, which is the result of the best fitness among the 300 sample datasets, is 12.96%. The results are promising and they are presented in Table 11.

Conclusions

ECMC is a very difficult problem so that very little research on computer-aided optimization for ECMC can be found in the existing literature. In this paper, a new framework called MIO LS-SVCM combining committee machine, MIO LS-SVM and evolutionary optimization for ECMC has been successfully developed and implemented. In the case of ECMC, the data consists of tunable and untunable parameters. It is unwise to include untunable parameters such as IAT, ET, BP, and BV for modelling and optimization. With reference to the idea of committee machine, this study firstly proposes to divide such complex system model into a number of simpler sub-models based on reasonable combinations of untunable parameters, so that the ECC sub-models do not contain any untunable parameter. With this division, the MIO LS-SVM in the framework can be employed to produce a set of ECC sub-models. Then, the models are systematically weighted and combined to form a multi-objective function for optimization. Finally, two common evolutionary optimization techniques, GA and PSO, are applied to the objective function to search for optimal values for calibration maps.

Evaluation tests show that the overall prediction accuracy of the LS-SVCM models is very good, at 94.1%. Experimental results also show that PSO is superior to GA under the LS-SVCM. Based on the optimal calibration maps obtained from PSO, the overall system performance relative to D_{best} is improved by 12.96%. Furthermore, the output performance using the recommended optimal maps is in good agreement with the engine real performance (about a 94.2% fit). Therefore, the proposed approach of MIO LS-SVCM + PSO is effective and performs well in this application. From the perspective of automotive engineering, this paper is the first in the literature that applied computational intelligence approach to solve the ECMC problem, and the proposed intelligent approach can greatly shorten the calibration time than the traditional manual approach.

In addition, the research is the first attempt at integrating diverse paradigms (Latin-hypercube sampling, nonlinear regression, MIO LS-SVCM and GA/PSO) into a general framework for large-scale constrained multivariable optimization problems under the conditions of unknown system model and the existence of untunable and tunable system parameters. The successful integration of GA and PSO with MIO LS-SVCM also indicates that the proposed modelling and optimization framework can fit many evolutionary optimization techniques. It is believed that the proposed framework can be applied to other similar engineering modelling and optimization problems as well.

TABLE 7 OPTIMAL COMPENSATION MAPS WITH THE GA

Optimal Fuel Compensation Map for ET

ET/℃	10	20	30	40	50	60	70	80	90	100	110
(% fuel trim)	33	32	26	20	17	14	7	3	0	3	5

Optimal Ignition Compensation Map for ET

ET/℃	10	20	30	40	50	60	70	80	90	100	110
(Ignition angle trim)	3.5	3	2.5	2	1.5	1	0.5	0	0	-2	-3.5

Optimal Fuel Compensation Map for IAT

IAT/℃	10	20	30	40	50	60
(% fuel trim)	5	3.5	2	0	-2	-3

Optimal Ignition Compensation Map for IAT

IAT/℃	10	20	30	40	50	60
(Ignition angle trim)	-6	-4.5	-2	0	-4	-7

Optimal Fuel Compensation Map for BV

BV/V	11	12	13	14
(μsec)	1153	1062	937	850

Optimal Fuel Compensation Map for cranking

Cranking time/ms	0-200
(% trim)	481

TABLE 8 PSO PARAMETERS

Number of generation	1000
Population size	50
Inertial weight	0.9
Cognitive parameter	2
Social parameter	2

TABLE 9 PSO RESULTS OF DIFFERENT OUTPUT FUNCTIONS

Engine output function $M_{h,i}(x)$	$Knk_i(x)$(%)	$CT_i(x)$(s)	$\Delta\lambda_i(x)$	$\Delta T_i(x)$(Nm)
	h=1	*h=2*	*h=3*	*h=4*
$M_{h,20,20,12}(x)$	15.77	0.33	0.08	-0.08
$M_{h,30,20,12}(x)$	15.85	3.06	0.08	-0.67
$M_{h,40,20,12}(x)$	16.04	1.55	0.07	6.41
$M_{h,50,20,12}(x)$	15.89	1.14	0.05	2.33
$M_{h,60,20,12}(x)$	15.90	3.76	0.05	-1.66
$M_{h,70,20,12}(x)$	16.39	0.51	0.07	0.68
$M_{h,50,30,12}(x)$	16.00	2.30	0.05	0.91
$M_{h,60,30,12}(x)$	16.00	2.09	0.06	0.27
$M_{h,70,30,12}(x)$	16.00	2.81	0.47	0.60
$M_{h,80,30,12}(x)$	15.77	2.34	0.48	2.09
$M_{h,90,30,12}(x)$	16.35	1.45	0.79	1.92
$M_{h,100,30,12}(x)$	16.28	0.80	0.50	0.27
$M_{h,90,40,12}(x)$	16.39	1.70	0.06	0.01
$M_{h,100,40,12}(x)$	15.85	2.46	0.96	1.10
$M_{h,100,50,12}(x)$	16.22	2.02	0.15	0.68
$M_{h,20,20,13}(x)$	15.87	1.13	0.11	0.11
$M_{h,30,20,13}(x)$	15.95	2.06	0.11	-0.47
$M_{h,40,20,13}(x)$	16.14	0.55	0.01	6.66
$M_{h,50,20,13}(x)$	15.99	1.14	0.03	2.53
$M_{h,60,20,13}(x)$	16.00	1.76	0.08	-1.46
$M_{h,70,20,13}(x)$	16.49	0.51	0.10	0.88
$M_{h,50,30,13}(x)$	16.10	1.30	0.07	1.11
$M_{h,60,30,13}(x)$	16.10	1.01	0.09	0.47
$M_{h,70,30,13}(x)$	16.10	2.81	0.50	0.81
$M_{h,80,30,13}(x)$	15.70	0.79	0.32	1.89
$M_{h,90,30,13}(x)$	16.13	1.56	0.15	2.12
$M_{h,100,30,13}(x)$	16.38	0.85	0.53	-1.46
$M_{h,90,40,13}(x)$	16.50	1.70	0.09	0.20
$M_{h,100,40,13}(x)$	16.71	1.19	0.13	2.27
$M_{h,100,50,13}(x)$	16.32	2.02	0.17	0.88
Overall average	16.11	1.62	0.22	1.05
Fitness value		-33.7		

TABLE 10 OPTIMAL COMPENSATION MAPS WITH PSO

Optimal Fuel Compensation Map for ET

ET/℃	10	20	30	40	50	60	70	80	90	100	110
(% fuel trim)	34	32	27	22	16	13	7	2	0	3	6

Optimal Ignition Compensation Map for ET

ET/℃	10	20	30	40	50	60	70	80	90	100	110
(Ignition angle trim)	3.5	3	2.5	2	1.5	1	0.5	0	0	-2	-4

Optimal Fuel Compensation Map for IAT

IAT/℃	10	20	30	40	50	60
(% fuel trim)	7	5	2	0	-2	-5

Optimal Ignition Compensation Map for IAT

AT/℃	10	20	30	40	50	60
(Ignition angle trim)	-6	-4	-2	0	-3	-5

Optimal Fuel Compensation Map for BV

BV/V	11	12	13	14
(μsec)	1124	1025	895	790

Optimal Fuel Compensation Map for cranking

Cranking time/ms	0-200
(% trim)	469

TABLE 11 COMPARISON BETWEEN OPTIMIZATION RESULTS, ACTUAL TEST RESULTS AND D_{best}

	$\overline{knk}$ (%)	$\overline{CT}$ (s)	$\overline{\Delta\lambda}$	$\overline{\Delta T}$ (Nm)	Overall
Actual test results using the set-points recommended by PSO, D_a	16.21	1.50	0.24	1.12	-
Optimization results using PSO, O_p	16.11	1.62	0.22	1.05	-
Optimization results using GA, O_g	16.42	1.75	0.52	1.02	-
Optimization results using the best set-points among the 300 sample datasets, D_{best}	18.41	1.80	0.28	1.23	-
Accuracy (O_p relative to D_a) $(1-\frac{\lvert O_p - D_a \rvert}{D_a})\times 100\%$	99.38%	92.00%	91.67%	93.75%	94.20%
Actual improvement (D_a relative to D_{best}) $\frac{D_{best}-D_a}{D_{best}}\times 100\%$	11.95%	16.67%	14.29%	8.94%	12.96%

ACKNOWLEDGMENT

The research is supported by the University of Macau Research Grant MYRG149(Y2-L2)-FST11-WPK. The author would also like to thank the support from Dr. Chi Man Vong and Dr. Weng Fai Ip of Faculty of Science and Technology, University of Macau. Finally, the contribution of Ms. Siyu Jia is also appreciated.

REFERENCES

Bell, A.G. "Four-stroke Performance Tuning: A Practical Guide." 3rd edn., Haynes Publishing Group, 2006.

Chen, S., Wang, W., & Van Zuylen, H. "Construct Support Vector Machine Ensemble to Detect Traffic Incident." Expert Systems with Applications, 36, 8, 10976-10986, 2009.

Celik, V. & Arcaklioglu, E. "Performance Maps of a Diesel Engine." Applied Energy, 81, 247-259, 2005.

Chiou, J.S., Liu, M.T. "Using Fuzzy logic Controller and Evolutionary Genetic Algorithm for Automotive Active Suspension System." International Journal of Automotive Technology, 10, 6, 703-710, 2009.

Clerc, M. & Kennedy, J. "The Particle Swarm-explosion, Stability, and Convergence in a Multidimensional Complex Space." IEEE Transactions on Evolutionary Computation, 6, 58-73, 2002.

Dickinson, P. & Shenton, A.T. "Dynamic Calibration of Fuelling in the PFI SI Engine." Control Engineering Practice, 17, 26-38, 2009.

Errico, G. D., Cerri, T. & Pertusi, G. "Multi-objective Optimization of Internal Combustion Engine by Means of 1D Fluid-dynamic Models." Applied Energy, 88, 767-777, 2011.

Garcia-Nieto, S., Martinez, M., Blasco, X. & Sanchis, J. "Nonlinear Predictive Control Based on Local Model Networks for Air Management in Diesel Engines." Control Engineering Practice, 16, 1399-1413, 2008.

Gauchia, A., Diaz, V., Boada, M.J.L., Boada, B.L. "Torsional Stiffness and Weight Optimization of a Real Bus Structure." International Journal of Automotive Technology, 11, 1, 41-47, 2010.

Hartman, J. "Fuel Injection Installation, Performance Tuning, Modifications." Motorbooks International, 1993.

Haykin, S. "Neural Networks: A Comprehensive Foundation." 2nd edn., Prentice Hall, 1999.

Jurgen, R. "Automotive Electronics Handbook." 1st edn., McGraw-Hill Press, 1995.

Kennedy, J., & Eberhart, R.C. "Particle Swarm Optimization." Proceedings of International Conference on Neural Networks, 1942-1948, 1995.

Krogh, A., & Vedelsby, J. "Neural Network Ensembles, Cross Validation and Active Learning." Advances in Neural Information Processing Systems, 7, 231-238, 1995.

Lunani, M., Sudjianto, A., & Johnson, P.L. "Generating Efficient Training Samples for Neural Networks Using Latin Hypercube Sampling." Proceedings of ANNIE'95 Artificial Neural Networks in Engineering, 209-214, 1995.

Li, G.Y. "Application of Intelligent Control and MATLAB to Electronically Controlled Engines (In Chinese)." Publishing House of Electronics Industry, 2007.

MoTeC. "MoTeC M800/M880 User's Manual." MoTeC Pty Ltd, 2002.

Pyle, D. "Data Preparation for Data Mining." 1st edn., Morgan Kaufmann Press, 1999.

Qi, K. P., Feng, L. Y., Leng, X. Y., Du, B. G. & Long, W. Q. "Simulation of Quasi-dimensional Combustion Model for Predicting Diesel Engine Performance." Applied Mathematical Modelling, 35, 930-940, 2011.

Rask, E., & Sellnau, M. "Simulation-Based Engine Calibration: Tools, Techniques, and Applications." SAE Paper No. 2004-01-1264, 2004.

Robert Bosch GmbH Staff. "Gasoline-Engine Management." 2nd edn. Robert Bosch GmbH and Professional Engineering Publishing Limited, 2004.

Suykens, J., Gestel, T., Brabanter, J., Moor, B., & Vandewalle, J. "Least Squares Support Vector Machines. World Scientific." 2002.

Saerens, B., Vandersteen, J., Persoons, T., Swevers, J., Diehl, M., & Van den Bulck, E. "Minimization of the Fuel Consumption of a Gasoline Engine using Dynamic Optimization." Applied Energy, 86, 1582-1588, 2009.

Trelea, I.C. "The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection." Information Processing Letters, 85, 317-325, 2003.

Tan, L. G., Gong, J. K., Zhai, L. Q. & Liu, J. W. "Modeling and Simulation of Fuel-injection MAP of Electronically Controlled Engine." Automotive Engineering, 28, 630-633 (In Chinese), 2006.

Togun, N. K. & Baysec, S. "Prediction of Torque and Specific Fuel Consumption of a Gasoline Engine by using Artificial Neural Networks." Applied Energy, 87, 349-355, 2010.

Vong, C.M., Wong, P.K. & Li, Y.P. "Predication of Automotive Engine Power and Torque using Least Squares Support Vector Machines and Bayesian Inference." Engineering Applications of Artificial Intelligence, 19, 3, pp. 277-287, 2006.

Wang, S.S., Li, L.H., Li, J.W., & Dong, H.G. "Principles and Service of Automotive Electronic Control Systems." 3rd edn., Beijing Institute of Technology Press (In Chinese), 2006.

Wong, P.K., Tam, L.M., Li, K., & Wong, H.C. "Automotive Engine Idle Speed Control Optimization using Least-squares Support Vector Machines and Genetic Algorithm." International Journal of Intelligent Computing and Cybernetics, 1, 598-616, 2008.

Wong, P.K., Vong, C.M., Tam, L.M., & Li, K. "Data Preprocessing and Modelling of Electronically-controlled Automotive Engine Power Performance using Kernel Principal Components Analysis and Least-square Support Vector Machines." International Journal of Vehicle Systems, Modelling and Testing, 3, 4, 312-330, 2008.

Wu, B. & Filipi, Z., Prucka, R., Kramer, D., & Ohl, G. "A Simulation-based Approach for Developing Optimal Calibrations for Engines with Variable Valve Actuation." Oil & Gas Science and Technology- Revue de l'IFP, 62, 539-553, 2007.

Zhang, J. Y., Shen, T. L. & Marino, R. "Model-based Cold-start Speed Control Scheme for Spark Ignition Engines." Control Engineering Practice, 18, 1285-1294, 2010.

List of Abbreviation

BP	Barometric pressure
BV	Battery voltage
CT	Cranking time
DOE	Design of experiment
ECC	Engine compensation control
ECCS	Engine compensation control system
ECMC	Engine compensation map calibration
ET	Engine temperature
EMS	Engine management system
GA	Genetic algorithm
IAT	Inlet air temperature
LS-SVM	Least-squares support vector machine
LS-SVCM	Least-squares support vector committee machine
MIO	Multi-input/output
NN	Neural network
PSO	Particle swarm optimization

Pak Kin Wong received the Ph.D. degree in Mechanical Engineering from The Hong Kong Polytechnic University, Hong Kong, China, in 1997. He is currently the Head of the Department of Electromechanical Engineering, Faculty of Science and Technology, University of Macau. His research interests include automotive engineering, fluid transmission and control, engineering applications of artificial intelligence, and mechanical vibration. He has published over 100 scientific papers in refereed journals, book chapters, and conference proceedings.